EXERCISE-12

Intro to Constraints; NOT NULL and UNIQUE Constraints.

Global Fast Foods has been very successful this past year and has opened several new stores. They need to add a table to their database to store information about each of their store's locations. The owners want to make sure that all entries have an identification number, date opened, address, and city and that no other entry in the table can have the same email address. Based on this information, answer the following questions about the global_locations table. Use the table for your answers.

Global Fast Foods global_locations Table						
NAME	TYPE	LENGTH	PRECISION	SCALE	NULLABLE	DEFAULT
id	pk				No	
name						
date_opened					No	
address					No	
city					No	
zip_postal_code						

phone						
email	**uk**					
manager_id						
emergency_ contact						

1. What is a “constraint” as it relates to data integrity?

Database can be as reliable as the data in it, and database rules are implemented as Constraint to maintain data integrity. For example these constraints may prohibit deletion of a table or some row when insertion, updation or deletion is executed. Type of constraints:

· **PRIMARY KEY Constraint**

· **UNIQUE Constraint**

· **FOREIGN KEY Constraint**

· **CHECK Constraint with condition applied on the column/columns (they work at row level)**

· **NOT NULL Constraint (implemented at row level using special CHECK Constraint having condition IS NOT NULL for single column)**

2. What are the limitations of constraints that may be applied at the column level and at the table level?

· **Constraints referring to more than one column are defined at Table Level**

· **NOT NULL constraint must be defined at column level as per ANSI/ISO SQL standard.**

· **If word CONSTRAINT is used in a CREATE TABLE statement, I must specify constraint name. Also, that is why, Table level constraint must be user-named.**

3. Why is it important to give meaningful names to constraints?

· **If a constraint is violated in a SQL statement execution, it is easy to identify the cause with user-named constraints.**

· **It is easy to alter names/drop constraint.**

· **Handling production issues may be faster with user-named constraints**

4. Based on the information provided by the owners, choose a datatype for each column. Indicate the length, precision, and scale for each NUMBER datatype.

Global Fast Foods global_locations Table						
NAME	TYPE	DataType	LENGTH	PRECISION	SCALE	NULLABLE
id	pk	**NUMBER**	**6**	**0**		No
name		**VARCHAR2**	**50**			
date_opened		**DATE**				No
address		**VARCHAR2**	**50**			No
city		**VARCHAR2**	**30**			No
zip_postal_code		**VARCHAR2**	**12**			
phone		**VARCHAR2**	**20**			
email	uk	**VARCHAR2**	**75**			
manager_id		**NUMBER**	**6**	**0**		
emergency_contact		**VARCHAR2**	**20**			

5. Use “nullable” to indicate those columns that can have null values.

Global Fast Foods global_locations Table						
NAME	TYPE	DataType	LENGTH	PRECISION	SCALE	NULLABLE
id	pk	NUMBER	6	0		**No**
name		VARCHAR2	50			**Yes**
date_opened		DATE				**No**
address		VARCHAR2	50			**No**
city		VARCHAR2	30			**No**
zip_postal_code		VARCHAR2	12			**Yes**
phone		VARCHAR2	20			**Yes**
email	uk	VARCHAR2	75			**Yes**
manager_id		NUMBER	6	0		**Yes**
emergency_contact		VARCHAR2	20			**Yes**

6. Write the CREATE TABLE statement for the Global Fast Foods locations table to define the constraints at the column level.

```
1   CREATE TABLE  f_global_locations
2   ( id NUMBER(6,0) CONSTRAINT f_gln_id_pk PRIMARY KEY ,
3   name VARCHAR2(50),
4   date_opened DATE CONSTRAINT f_gln_dt_opened_nn NOT NULL ENABLE,
5   address VARCHAR2(50) CONSTRAINT f_gln_add_nn NOT NULL ENABLE,
6   city VARCHAR2(30)  CONSTRAINT f_gln_city_nn NOT NULL ENABLE,
7   zip_postal_code VARCHAR2(12),
8   phone VARCHAR2(20),
9   email VARCHAR2(75) CONSTRAINT f_gln_email_uk UNIQUE,
10  manager_id NUMBER(6,0),
11  emergency_contact VARCHAR2(20)
12  );
13  DESCRIBE f_global_locations;
14
```

7. Execute the CREATE TABLE statement in Oracle Application Express.

Table created.

8. Execute a DESCRIBE command to view the Table Summary information.

DESCRIBE f_global_locations;

F_GLOBAL_LOCATIONS	ID	NUMBER	-	6	0	1	-	-	-
	NAME	VARCHAR2	50	-	-	-	✓	-	-
	DATE_OPENED	DATE	7	-	-	-	-	-	-
	ADDRESS	VARCHAR2	50	-	-	-	-	-	-
	CITY	VARCHAR2	30	-	-	-	-	-	-
	ZIP_POSTAL_CODE	VARCHAR2	12	-	-	-	✓	-	-
	PHONE	VARCHAR2	20	-	-	-	✓	-	-
	EMAIL	VARCHAR2	75	-	-	-	✓	-	-
	MANAGER_ID	NUMBER	-	6	0	-	✓	-	-
	EMERGENCY_CONTACT	VARCHAR2	20	-	-	-	✓	-	-

9. Rewrite the CREATE TABLE statement for the Global Fast Foods locations table to define the UNIQUE constraints at the table level. Do not execute this statement.

CREATE TABLE f_global_locations (id NUMBER(6,0) CONSTRAINT f_gln_id_pk PRIMARY KEY ,name VARCHAR2(50),date_opened DATE CONSTRAINT f_gln_dt_opened_nn NOT NULL ENABLE, address VARCHAR2(50) CONSTRAINT f_gln_add_nn NOT NULL ENABLE, city VARCHAR2(30) CONSTRAINT f_gln_city_nn NOT NULL ENABLE, zip_postal_code VARCHAR2(12),phone VARCHAR2(20),**email VARCHAR2(75) ,** manager_id NUMBER(6,0), emergency_contact VARCHAR2(20), **CONSTRAINT f_gln_email_uk UNIQUE(email)**);